



11-14-06

AF

U.S. EXPRESS MAIL "POST OFFICE TO ADDRESSEE" SERVICE
DEPOSIT INFORMATIONExpress Mail Label No.: EV 906810876 USDate of Deposit: November 13, 2006B R I N K S
H O F E R
G I L S O N
& L I O N E

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Appln. of: George V. Guyan et al.

Appln. No.: 09/305,146

Filed: May 4, 1999

For: COMPONENT BASED INFORMATION
LINKING DURING CLAIM PROCESSING

Attorney Docket No: 10022/252-1

Examiner: Rimell, S.

Art Unit: 2175

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

TRANSMITTAL

Sir:

Attached is/are:

Amended Appeal Brief (original and 1 copy), Response to Notification of Non-Compliant Appeal Brief.
 Return Receipt Postcard.

Fee calculation:

No additional fee is required.
 Small Entity.
 An extension fee in an amount of \$____ for a ____-month extension of time under 37 C.F.R. § 1.136(a).
 A petition or processing fee in an amount of \$____ under 37 C.F.R. § 1.17(____).
 An additional filing fee has been calculated as shown below:

	Claims Remaining After Amendment		Highest No. Previously Paid For	Present Extra	Small Entity		Not a Small Entity		
					Rate	Add'l Fee	or	Rate	Add'l Fee
Total		Minus			x \$25=			x \$50=	
Indep.		Minus			X100=			x \$200=	
First Presentation of Multiple Dep. Claim					+\$180=			+\$360=	
					Total	\$	Total	\$	

Fee payment:

A check in the amount of \$____ is enclosed.
 Please charge Deposit Account No. 23-1925 in the amount of \$____. A copy of this Transmittal is enclosed for this purpose.
 Payment by credit card in the amount of \$____ (Form PTO-2038 is attached).
 The Director is hereby authorized to charge payment of any additional filing fees required under 37 CFR § 1.16 and any patent application processing fees under 37 CFR § 1.17 associated with this paper (including any extension fee required to ensure that this paper is timely filed), or to credit any overpayment, to Deposit Account No. 23-1925.

Respectfully submitted,

John C. Freeman, Esq. (Reg. No. 34,483)

November 13, 2006

Date

BRINKS HOFER GILSON & LIONE
NBC Tower – Suite 3600, 455 N. Cityfront Plaza Drive, Chicago, IL 60611-5599



"Express Mail" mailing label number EV 906810876 US

Date of Deposit: November 13, 2006

PATENT
CASE NO. 10022/252-1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application:)
George V. Guyan et al.)
Serial No.: 09/305,146) Group Art Unit: 2175
Filed: May 4, 1999)
For: COMPONENT BASED)
INFORMATION LINKING)
DURING CLAIM)
PROCESSING)

RESPONSE TO NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

On October 13, 2006 a Notification of Non-Compliant Appeal Brief ("the Notification") was mailed in which it was asserted that Appellants' Appeal Brief filed on July 24, 2006 did not contain a concise explanation of the subject matter defined in each of the independent claims. The Notification stated that "[t]he Summary of Claimed Subject Matter should be a concise discussion of the content of each independent claim, rather than discussing the invention in general. The Summary should be correlated to each independent claim." Appellants traverse the Notification in that the explanation of the independent claims in Appellants' Appeal Brief is concise. This is evidenced by the above statement in the Notification which does not provide any specifics as to how the

explanation is not concise and how to correct the explanation. This is unfair to Appellants in that it requires them to guess the reasoning behind the Notification and how to respond.

The Notification further objected to the Status of Amendments Section of Appellants' Appeal Brief for providing the status of Amendments After Final filed prior to the mailing of the Final Office Action on February 24, 2006. Appellants traverse the objection in that MPEP § 1205.02 states that the Section is to include "[a] statement of the status of any amendment filed subsequent to final rejection" (emphasis supplied).

Despite the impropriety of the objections, an amended Appeal Brief is being filed concurrently with the present Response to replace the Appeal Brief filed on July 24, 2006 in accordance with MPEP § 1205.03.

Respectfully submitted,



John C. Freeman
Registration No. 34,483
Attorney for Appellants

BRINKS HOFER
GILSON & LIONE
P.O. Box 10395
Chicago, Illinois 60610
(312) 321-4200

Dated: November 13, 2006



"Express Mail" mailing label number EV 906810876 US
Date of Deposit: November 13, 2006

PATENT
CASE NO. 10022/252-1

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application:)
George V. Guyan et al.)
Serial No.: 09/305,146) Group Art Unit: 2175
Filed: May 4, 1999) Examiner: Rimell, S.
For: COMPONENT BASED)
INFORMATION LINKING)
DURING CLAIM)
PROCESSING)

AMENDED APPEAL BRIEF

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This Appeal is in response to the Final Office Action mailed February 24, 2006 and the Notice of Panel Decision from Pre-Appeal Brief Review mailed on June 22, 2006¹.

¹ On July 24, 2006, Appellants filed an Appeal Brief. A Notice of Non-Compliant Appeal Brief was mailed on October 13, 2006. Since the present Appeal Brief is being filed within one month of the mailing of the Notice of Non-Compliant Appeal Brief, the present Appeal Brief is timely filed.

I. REAL PARTY IN INTEREST

It is believed that Accenture LLP is the real party of interest in this Appeal pursuant to the following: 1) recorded assignments of the above-identified application to AC Properties B.V. by both of the inventors of record, 2) a recorded assignment of the above-identified application to Andersen Consulting LLP and 3) a Change of Name document filed on August 16, 2001 to be recorded² that demonstrates that Andersen Consulting LLP legally changed its name to Accenture LLP.

II. RELATED APPEALS AND INTERFERENCES

The undersigned, John C. Freeman, is not aware of any other appeals, interferences or other judicial proceedings that may be related to, would directly affect or be directly affected by or have a bearing on the Board's decision in the pending Appeal.

III. STATUS OF CLAIMS

The status of the claims is as follows:

Claims 1-21 and 23-40 are canceled.

Claims 22 and 41-65 are finally rejected under 35 U.S.C. § 112, first paragraph, for failing the written description requirement.

Claims 66 and 67 are finally rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 5,950,169 to Borghesi et al.

² As of the date of the present Appeal Brief, the U.S. Patent Office has not responded to Appellants' request to record the document. Steps will be taken to obtain a response.

The previously mentioned rejections of claims 22 and 41-67 are the subject of this Appeal.

IV. STATUS OF AMENDMENTS

A Final Office Action was mailed on February 24, 2006. No Amendments have been filed from February 24, 2006 to the filing of the present Appeal Brief. A Pre-Appeal Brief Request for Review was filed on April 24, 2006. No other papers have been filed from February 24, 2006 to the filing of the present Appeal Brief.

V. SUMMARY OF CLAIMED SUBJECT MATTER

An understanding of the inventions of independent claims 22, 66 and 67 can be made upon a review of the embodiments of the inventions shown in FIGS. 2A and 13-15 of the specification. Note that in the description to follow, like elements will employ identical identification numerals. FIG. 2A shows an embodiment of a server based framework utilizing component based architecture that includes an Architecture Object 200, an Application Object 202, a User Interface Form 204, a User Interface Controller 206, a Client Component Adapter 208, a COM Component Interface 210, and a Server Component 222 (P. 16, II. 11-16). In general, each component of the server based framework of FIG. 2A stores data in an object of the component (P. 16, II. 18-19). Functions which manipulate the object are encapsulated with the object data (P. 16, I. 20). Later, the object data stored in one component of the server based framework can be manipulated by other components of the server based framework by utilizing the encapsulated functions (P. 16, II. 21-22).

The Architecture Object 200 is responsible for providing all client architecture services (i.e., codes table access, error logging, etc.), and a single point of entry for

architecture services. The Architecture Object 200 is also responsible for allowing the architecture to exist as an autonomous unit, thus allowing internal changes to be made to the architecture with minimal impact to application. The Architecture Object 200 provides a code manager, client profile, text manager, ID manager, registry manager, log manager, error manager, and a security manager.

The Application Object 202 has a method to initiate each business operation in the application (P. 17, II. 23-24). The Application Object 202 is responsible for instantiating each User Interface (hereinafter “UI”) Controller 206, passing data/business context to the target UI Controller 206, and invoking standard services such as initialize controller, initializing Form and Initialize Architecture (P. 18, II. 6-10).

The UI form 204’s primary responsibility is to forward important events to its UI controller 206 (P. 18, II. 13-14). The UI form 204 presents an easy-to-use, graphical interface to the user and informs its UI controller 206 of important user actions (P. 18, II. 22-23).

The UI Controller 206 contains the majority of logic to manipulate Business Objects 207 and manage the appearance of its UI form 204 (P. 19, II. 1-2). The UI Controller 206 is responsible for handling events generated by the user interacting with the UI form 204 (P. 19, II. 20-21).

The Business Object (BO) 207’s primary functionality is to act as a data holder, allowing data to be shared across User Interface Controllers 206 using an object-based programming model (P. 19, II. 29-31).

Client Component Adapters (CCAs) 208 are responsible for retrieving, adding, updating, and deleting Business Objects 207 in the database (P. 20, II. 21-22). As

shown in FIG. 2A, the Client Component Adapter 208 is in communication with a COM Component Interface (CCI) 210. As denoted by the dashed lines of FIG. 2A, the CCI 210 is not a physical entity. Instead, it represents a “contract” for services to be provided by a component (P. 21, II. 2-3).

As shown in FIG. 2A, the CCI is in communication with a Server Component 222. The Server Components 222 encapsulate all access to the database, carrying out data access operations and define business transaction boundaries (P. 21, II. 24-25 and P. 52, II. 40-41). In addition, the Server Component 222 is responsible for ensuring that business rules are honored during data access operations (P. 21, II. 25-27 and P. 52, II. 40-41). The Server Component 222 performs data access operations on behalf of CCAs 208 or other components and participates in transactions spanning server components 222 by communicating with other server components 222 (P. 21, II. 29-31). A server component 222 often collaborates with other server components to complete a business transaction (P. 53, II. 7-8)).

As shown in FIG. 14, a Claim Folder 1402 is in communication with an Event Processor 1400. The Claim Folder 1402 manages claim information from first notice through closing and archiving (P. 138, II. 14-15). The Claim Folder 1402 manages claim information by providing a structured and easy to use interface that supports multiple business processes for handling claims (P. 138, II. 15-16).

The Claim Folder 1402 supports several primary processes, such as first notice of loss, claim inquiry, initiation of claim handling, investigation and evaluation of a claim, identifying claim events and managing the physical file corresponding to the claim (P.

138, I. 25 – P. 139, I. 21). The folder design allows quick access to various levels of information within the claim for many different reasons (P. 139, II. 2-3).

The Claim Folder 1402 decomposes a claim into different levels that reflect the policy, the insured, the claim, the claimants, and the claimant's lines (P. 141, II. 1-2). The claim tree in the Claim Folder window decomposes the claim into various levels depending on the specific composition of the claim (P. 143, II. 7-9). For example, the levels can include a policy level, a claim level, a participant level and a line level (P. 141, II. 15-26). Each level has a structured set of information that applies to it (P. 141, II. 2-3). For example, the claim level of the claim has information on the claim status, line of business, and performers (P. 141, II. 3-4). An individual line level has information which includes the line type, jurisdiction, and property or vehicle damages (P. 141, II. 4-6). The claimant level contains contact information as well as injury descriptions (P. 141, II. 6-7). The Claim Folder 1402 can be in view mode or edit mode for a specific level in the claim tree (P. 144, II. 25-26). A tool bar for the Claim Folder 1402 is used to perform a number of functions available to the user, such as viewing and editing the contents of the Claim Folder 1402 (Pages 145-157).

Events are triggered in the Claim Folder 1402 by performing certain actions like changing a jurisdiction, identifying an injury, or closing a line (P. 141, II. 28-29). Other general events are triggered in the Event Section on most levels by clicking the one that has occurred (P. 141, II. 29-31). These events are processed by the Event Processor 1400 and could generate any number of responses, such as triggering new tasks in the Task Assistant 1402 for a claim (P. 141, I. 31 – P. 142, I. 2).

As shown in FIG. 14, the Event Processor 1400 is in communication with the Claim Folder 1402. The Event Processor 1400 utilizes a common queue 208 of events 1006 that are populated by any component 1402 of the system to identify what events have occurred (P. 185, II. 9-11). Working this queue, the Event Processor 1400 determines the appropriate response for an event and provides information to other components that need to process them (P. 185, II. 11-13). The Event Processor 1400 does not process any events itself and maintains clear encapsulation of system responsibilities (P. 185, II. 13-15). Encapsulation enforces data abstraction through the organization of data into small, independent objects that can communicate with each other (P. 9, II. 19-20). Encapsulation protects the data in an object from accidental damage, but allows other objects to interact with that data by calling the object's member functions and structures (P. 9, II. 21-23). For example, an event that affects claim data is processed by the claim component (P. 185, II. 15-16). The Event Processor 1400 works behind the scenes of all claims applications to listen for significant events that have occurred in the life of various entities in the system like claims (but potentially many more like accounts or policies in the future) (P. 183, II. 19-22). It determines what the response should be to each event and passes it onto the system component that will process it (P. 183, II. 22-23).

As shown in FIGS. 14 and 15, the Event Processor 1400 is in communication with a Task Engine 1404, which in turn runs on the server 222 (P. 137, II. 29-31). The Task Engine 1404 processes the most common set of event responses, those that need to generate tasks 1406 based on events 1006 that have occurred (P. 183, II. 29-30). It compares the tasks that have been defined to the system to a set of claim criteria to tell

which tasks should be added and which tasks should now be marked complete (P. 184, II. 1-3). The Task Engine 1404 follows a process of evaluating events 1006, determining claim characteristics, and matching the claim's characteristics to tasks defined in the Task Library 1500, which is discussed below (P. 185, II. 18-20).

As shown in FIGS. 14 and 15, the Task Assistant 1406 receives tasks 1406 from the Task Engine 1404. The Task Assistant 1406 is the cornerstone of a claim professional's working environment (P. 179, II. 8-9). It provides diary functions at a work step level that allow the management of complex claim events (P. 179, II. 9-10). It enables the consistent execution of claim best practices by assembling and re-assembling all of the tasks that need to be performed for a claim based on detailed claim characteristics that come from regulatory compliance requirements, account servicing commitments, and best practices for handling all types of claims (P. 179, II. 10-14).

Within the Task Assistant 1402, claim professionals have the ultimate control to determine if and when tasks need to be completed (P. 180, II. 4-5). They also have the ability to add tasks to the list to represent work they do that is not reflected in standard definitions of tasks in the system (P. 180, II. 5-7).

While claim professionals are the primary users of the Task Assistant 1402, others use the application as well (P. 181, II. 2-3). The entire claims department utilizes the Task Assistant 1402 to structure work and communicate with one another (P. 181, II. 3-4). Team leaders use the Task Assistant 1402 to conduct file review and to guide the work of the claim professional (P. 181, II. 4-5). Administrative staff uses the Task Assistant 1402 as a means to receive work and to communicate the completion of that

work (P. 181, II. 6-7). Claim professionals use the Task Assistant 1402 to complete work and to request assistance from team leaders and specialty claim professionals (P. 181, II. 7-9).

The Task Assistant 1402 requires a new type of user to set-up and maintain the variety of tasks that are created (P. 181, II. 11-12). As shown in FIG. 15, a Task Librarian 1502 maintains a Task Library 1500, which contains the list of all the standardized tasks across the organization (P. 181, II. 12-13). The Task Librarian 1502 defines rules which cause tasks to be placed on task lists based on claim characteristics, dates which define when tasks are due, and task enablement through other applications (P. 181, II. 13-16). A key user interface for the Task Engine 1404 is the Task Library 1500 (P. 185, I. 22).

FIG. 13 is a flow diagram of the operations utilized by the Task Assistant 1402 (P. 181, II. 19-20). The processing of tasks through the Task Assistant 1402 comprises the lifecycle of the task from its creation to its completion or deletion (P. 181, II. 20-22). In the first operation 1300, the Task Engine 1404 provides tasks to the Task Assistant 1402 (P. 181, II. 22-23). In the second operation 1302, the Task Assistant 1402 then displays the list of tasks provided by the Task Engine 1404 (P. 181, II. 23-24). In the third operation 1304, the user is allowed to add tasks and edit tasks provided by the Task Engine 1404 (P. 181, II. 24-25). The fourth operation 1306 occurs as the claim is processed and involves having the user and the Task Engine 1404 determine when the various tasks are completed (P. 181, II. 25-27). When a task is completed, the fifth operation 1308 occurs resulting in the generation of a historical record for any tasks which is determined to be completed (P. 181, II. 27-30).

Behind the functioning of the Task Assistant 1402, the Task Engine 1404 continually evaluates messages sent from other components and determines based on the rules established by the Task Librarian 1502, which tasks should be populated on the Task Assistant 1402 (P. 182, ll. 21-24). Messages are sent to the Task Assistant 1402 when something significant occurs in another component (P. 182, ll. 24-25). The messages contain the characteristics the Task Engine needs to evaluate in order to place the proper tasks on the task list (P. 182, ll. 25-27).

With the above summary in mind, claim 22 claims the invention as a system for displaying information about an insurance claim for an insured event. The system includes a server component having an event processor and a task engine application program that interacts with the event processor to enable the insurance claim to be processed. An example of such a server component can be found from the server component 222 shown in FIGS. 2A and 8 (P. 137, ll. 29-31). Claim 22 further includes a data component residing on the server component, the data component having a claim folder that decomposes a claim related to the insured event into a plurality of levels, the plurality of levels including a policy level, a claim level, a participant level and a line level. An example of such a data component can be found in the description of the claim folder 1402 at page 141, lines 1-7 and 15-26 and page 143, lines 7-9 of Appellants' Specification. Claim 22 clarifies that the server component is configured to generate a user interactive interface that interactively displays at least one of the plurality of levels reflecting information related to a policy, the claim, claimants and an insured person in a structured format to a plurality of users, and to allow each of the users to simultaneously interact with one of the plurality of levels to retrieve and enter

data for the same insurance claim. An example of such a configuration can be found in the Claim Folder 1402 of FIG. 14 with its view and edit modes as described at page 138, lines 15-16, page 144, ll. 25-26 and pages 145-147 of Appellants' Specification. Claim 22 further clarifies that the event processor maintains clear encapsulation of responsibilities of the system for displaying information from the event processor, wherein the responsibilities do not include functions performed by the event processor, interacts with the data component to identify a data event that affects data in the claim folder, determines a response, identifies a system component to enable the claim to be processed and transmits the data event to the identified system component. An example of such a configuration can be found in FIG. 14 and is described at page 9, lines 23-25 and page 185, lines 9-16 and 19-23 of Appellants' Specification. Claim 22 further clarifies that the identified system component is the task engine, the task engine evaluates the data event, determines claim characteristics and matches the characteristics to tasks to automatically generate a list of tasks to be taken by one of the plurality of users handling the insurance claim to direct a workflow for the insurance claim to be processed. An example of such a task engine can be found with the Task Engine 1404 of FIGS. 14 and 15 and described at page 137, lines 29-31, page 183, lines 29-30. page 184, lines 1-3 and page 185, lines 18-20 of Appellants' Specification.

Claim 66 claims the invention as a system that displays insurance claim about an insured event. The claimed system includes an event processor that identifies a data event, determines a response, identifies a system component to process an insurance claim and transmits information regarding the data event to the identified system component. An example of such an event processor is the Event Processor 1400

shown in FIGS. 14-15 and described at page 185, lines 9-16 and 19-23 of Appellants' Specification. Claim 66 further includes a task engine application program that interacts with the event processor to enable the insurance claim to be processed. An example of such a task engine application program can be found with the Task Engine 1404 of FIGS. 14 and 15 and described at page 137, lines 29-31, page 183, lines 29-30, page 184, lines 1-3 and page 185, lines 18-20 of Appellants' Specification. Claim 66 includes a data component having a claim folder that decomposes a claim related to the insured event into a plurality of levels, the plurality of levels including a policy level, a claim level, a participant level and a line level. An example of such a data component can be found in the description of the claim folder 1402 at page 141, lines 1-7 and 15-26 and page 143, lines 7-9 of Appellants' Specification. Claim 66 further includes a user interactive interface that is generated by a server that interactively displays information from at least one of the plurality of levels in a structured format to a plurality of users, allowing each of the users to simultaneously interact with one of the plurality of levels to retrieve and enter data for the same insurance claim, the entered data triggering the data event. An example of such a user interactive interface can be found in relation with the Claim Folder 1402 of FIG. 14 with its view and edit modes as described at page 138, lines 15-16, page 144, ll. 25-26 and pages 145-147 of Appellants' Specification. Claim 66 further clarifies that when the event processor identifies the task engine as the system component to process the insurance claim, the task engine evaluates the event, determines claim characteristics for the event and matches the characteristics to tasks to automatically generate a list of tasks to be taken by one of the plurality of users handling the insurance claim to direct a workflow for the insurance claim to be

processed. An example of such a task engine can be found with the Task Engine 1404 of FIGS. 14 and 15 and described at page 137, lines 29-31, page 183, lines 29-30. page 184, lines 1-3 and page 185, lines 18-20 of Appellants' Specification.

Claim 67 claims the invention as a system that displays insurance claim information. The claimed system includes a data component that includes a claim folder that decomposes a claim related to an insured event into a plurality of levels, the plurality of levels include a policy level, a claim level, a participant level and a line level. An example of such a data component can be found in the description of the claim folder 1402 at page 141, lines 1-7 and 15-26 and page 143, lines 7-9 of Appellants' Specification. Claim 67 includes a user interactive interface that is generated and interactively displays information from at least one of the plurality of levels in a structured format to a plurality of users, wherein a plurality of users via a plurality of interfaces is allowed to simultaneously interact with one of the plurality of levels to retrieve and enter data on the same insurance claim. An example of such a user interactive interface can be found in relation with the Claim Folder 1402 of FIG. 14 with its view and edit modes as described at page 138, lines 15-16, page 144, II. 25-26 and pages 145-147 of Appellants' Specification. Claim 67 further includes an event processor that identifies the entered data as a data event, determines a response for the data event and identifies a system component to process the response and transmits information for processing the claim to the identified system component. An example of such an event processor is the Event Processor 1400 shown in FIGS. 14-15 and described at page 185, lines 9-16 and 19-23 of Appellants' Specification.

There are no means-plus-function terms or step-plus-function terms in independent claims 22, 66, 67, which are argued separately below in Section VII.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

There are two grounds of rejection presented for review:

- 1) the rejection of claims 22 and 41-65 under 35 U.S.C. § 112, first paragraph, for failing the written description requirement; and
- 2) the rejection of claims 66 and 67 for being anticipated under 35 U.S.C. § 102(e) in view of Borghesi et al.

VII. ARGUMENT

A. 35 U.S.C. § 112, First Paragraph

Claims 22 and 41-65 are finally rejected under 35 U.S.C. § 112, first paragraph, for failing the written description requirement. In particular, claim 22 was rejected because the phrase “do not include functions performed by said event processor”, as added in Appellants’ Amendment of November 22, 2005, allegedly contained new matter. Appellants traverse the rejection. To provide some context, claim 22 was amended to clarify that the system responsibilities that are encapsulated by the event processor “do not include functions performed by said event processor.” An example of such encapsulated system responsibilities are disclosed in Appellants’ original Specification at page 185 which states:

The Event Processor does not process any events itself and maintains clear encapsulation of system responsibilities. For example, an event that affects claim data is processed by the claim component.

Appellants' original Specification at page 2 explains the concept of encapsulation as follows:

OOP [object oriented programming], therefore views a computer program as a collection of largely autonomous components, called objects, each of which is responsible for a specific task. This concept of packaging data, structures, and procedures together in one component or module is called encapsulation. (bracketed material added).

One of ordinary skill in the art would read the above two passages to mean that the embodiment of the event processor disclosed on page 185 and shown in FIG. 14 is compartmentalized in that it does not perform any of the functions regarding other components of the system, such as the components of the system for displaying information. Those functions are encapsulated in that they are to be performed only by the components for displaying information. One of ordinary skill would also understand encapsulation to mean that no functions of one component are mixed in with functions of another component. In the present case, if the event processor has some of its functions grouped with the functions that are performed by the components for displaying information, then it follows that there is a mixture of functions being delegated to the components and so encapsulation has not been achieved which is contrary to Appellants' own disclosure.

Further evidence that encapsulation does not allow for a mixture of functions is that to do so would lead to an inoperative device. For example, assume for arguments sake that encapsulation allows functions of the event processor to be grouped with the functions that are performed by the components for displaying information. Such a group of functions are then sent to the components for displaying information. In this

scenario, the grouped functions of the event processor will not be performed by the event processor since they are sent to the components for displaying information. Furthermore, the components for displaying information will not be able to perform the functions of the event processor sent with the other functions. Thus, the event processor is not fully functional in this scenario, which of course shows that the above assumption has no merit. Accordingly, one of ordinary skill would understand that encapsulation as described in Appellants' Specification does not allow for a mixture of functions and so the phrase "do not include functions performed by said event processor" does not involve new matter since it is merely a restatement that mixing of functions is not permitted for encapsulation.

It is noted that the Final Office Action bases its rejection on the fact that the offending amended language is not disclosed in the original specification. However, the Final Office Action has failed to take into account that a lack of literal basis in the specification for a negative limitation may not be sufficient by itself to establish a *prima facie* case for a lack of descriptive support. *Ex parte Parks*, 30 USPQ2d 1234, 1236 (Bd. Pat. App. & Inter. 1993), MPEP § 2173.05(i). The arguments above show that the Specification implies the negative limitation in question.

On a related matter, the last paragraph at page 2 of the Final Office Action asserts that Appellants' failure to assert that the offending amended language was not new matter was evidence that the language was new matter. There is no legal basis for such an assertion. First, there is no requirement that an assertion of no new matter be made. Second, Appellants previously amended claim 22 in their Amendment of May 31, 2005 to clarify that the event processor "maintains clear encapsulation from

responsibilities." Page 8 of the May 31st Amendment specified that the amendment was supported by at least page 185 of the specification. The Final Office Action mailed on August 23, 2005 did not dispute that there was support for the amendment. Since the offending amended language presented in Appellants' Amendment of November 22, 2005 was a rephrasing of the amended language of May 31, 2005 (which was not deemed to be new matter) and was not intended to change its intended scope or meaning, the offending amended language is not new matter. MPEP § 2163.07 I.

It is noted that claims 22 and 41-65 have not been rejected based on the prior art. Since the rejections of claims 22 and 41-65 have been shown to be improper, the claims should be allowed. Furthermore, the Final Office Action has not rejected the claims under 35 U.S.C. § 112, second paragraph, and so the Examiner has conceded that the claims are clear in meaning (see first paragraph of Remarks Section at page 4 of the Office Action) and that a prior art search of the claims was possible at the time of mailing of the Office Action. The absence of a rejection based on the prior art can only mean that the Examiner has not found any art that would render the claims unpatentable. See MPEP § 2163.06 I. Accordingly, there is no need to remand the case back to the Examiner for a further search.

B. 35 U.S.C. § 102

1. Claim 66

Claim 66 was finally rejected in the Final Office Action of February 24, 2006 under 35 U.S.C. § 102(e) as being anticipated by Borghesi et al. Appellants traverse this rejection. In particular, claim 66 recites a system that displays insurance claim information that includes a server that allows each of the users of a user interactive

interface “to simultaneously interact with one of the plurality of levels to retrieve and enter data for the same insurance claim.” The Final Office Action at page 4, lines 2-3 has asserted that “[a]ny one of the users at the computers (30 or 32 or 34) can interact with the levels of the claim folder to retrieve data of the folder and enter data into the folder.” The Final Office Action, for the first time, asserts at page 4, lines 14-16 that FIGS. 2-3 disclose that a claims folder is accessible to computer 30, 32, 34 via server 36. A review of FIGS. 2-3 does not disclose each of the users of computers 30, 32, 34 simultaneously interacting with a level to retrieve/enter data for an insurance claim. The Final Office Action also asserts at page 4, lines 16-17 that “in any client-server configuration, access to data by the client does not lock out access to that data by the other clients.” However, the Final Office Action has not cited one passage of Borghesi et al. that discloses such a client-server configuration. Indeed, it is apparent that the rejection is based on incorporating attributes to Borghesi et al. that simply are not explicitly or inherently present in the reference. This is improper. *Glaverbel S.A. v. Northlake Mkt'g & Supp., Inc.*, 45 F.3d 1550, 1554, 33 USPQ2d 1496, 1498 (Fed. Cir. 1995).

Borghesi et al. also fails to disclose matching determined claim characteristics to tasks “to automatically generate a list of tasks to be taken by any one of the plurality of users” as recited in claim 66. In other words, Borghesi et al. does not disclose a system that determines claim characteristics that are used to automatically generate a list of tasks. The Final Office Action at page 5 refers to the actions 220-230 shown in FIG. 8F as disclosing the recited list of tasks. However, FIG. 8F is merely a flow diagram showing a preferred workflow (Col. 3, ll. 49-50). Nowhere does Borghesi et al. disclose

that a list of items 220-230 shown in FIG. 8F is automatically generated. Furthermore, Borghesi et al. does not disclose automatically generating the list of items 220-230 based on a determination of claim characteristics. The Final Office Action asserts for the first time that:

FIGS. 8E through 8F illustrate that when a user selects the “create/edit” data function in FIG. 8E, this selection automatically generates a series of tasks that have to be performed according to the sequence defined in FIG. 8F. These steps are triggered by a determination of claim characteristics, such as defined by step 202. Step 220 can also be read as the identification of a claim characteristic. Identifying the characteristic then automatically triggers the generation of tasks, according to the define flow chart in FIG. 8F. This list of tasks is not suggested as being triggered manually or randomly, since the flow chart of FIG. 8F defines both the tasks themselves and the exact sequence in which they must be followed. (Final Office Action, page 4, l. 22, P. 5, l. 7)

The above assertion is imaginative to say the least. Borghesi et al. does not disclose that a user select a “create/data” function 202. Instead, item 202 is an information process that can be performed by a user, via a graphic user interface for example. (Col. 12, ll. 3-13). That process can include any of the processes shown in FIG. 8F. The processes are done voluntarily by a user by entering information via a keyboard. (Col. 12, ll. 14-36). There is nothing in Borghesi et al. that discloses the processes of FIG. 8F are automatically generated or generated by claim characteristics. Furthermore, no list of tasks is generated. FIG. 8F is merely a flow chart that helps the reader of the patent to understand processes that can be performed by a user. The flow chart is not itself generated by the system of Borghesi et al.

Finally, the Final Office Action asserts that the processes of FIG. 8F must be done automatically since it defines tasks and an “exact sequence in which they must be

followed.” The assertion has no merit. Using such logic would mean that a recipe for a food dish must be performed automatically since the recipe defines tasks and gives the exact sequence that the tasks must be performed. Furthermore, the assertion ignores Borghesi et al.’s own disclosure which states a user manually enters information via a keyboard for performing the processes of FIG. 8F. (Col. 12, ll. 14-16).

For the above reasons, the rejection of claim 66 is improper and should be withdrawn.

2. Claim 67

Claim 67 was finally rejected in the Final Office Action of February 24, 2006 under 35 U.S.C. § 102(e) as being anticipated by Borghesi et al. Appellants traverse this rejection. In particular, claim 67 recites a system that displays insurance claim information that includes a plurality of interfaces that allow a plurality of users “to simultaneously interact with one of the plurality of levels to retrieve and enter data on the same insurance claim.” The Final Office Action at page 4, lines 2-3 has asserted that “[a]ny one of the users at the computers (30 or 32 or 34) can interact with the levels of the claim folder to retrieve data of the folder and enter data into the folder.” The Final Office Action, for the first time, asserts at page 4, lines 14-16 that FIGS. 2-3 disclose that a claims folder is accessible to computer 30, 32, 34 via server 36. As stated above in Section VII B.1, the assertion has no merit. Accordingly, claim 67 is not anticipated by Borghesi et al.

Borghesi et al. also fails to disclose an event processor that “determines a response for the data event and identifies a system component to process the response and transmits information for processing the claim to the identified system component”

as recited in claim 67. The Final Office Action has failed to specifically identify any element in Borghesi et al. that corresponds to the above mentioned event processor. While page 4 of the Office Action mentions an event processor in conjunctions with FIG. 8E, the drawings do not show a processor. Evidently Borghesi et al. fails to disclose the recited event processor. Accordingly, Borghesi et al. does not anticipate claim 67. Note that Appellants also repeat their objections above in Section VII B.1. to those arguments in the Office Action regarding claim 66 which have been applied to claim 67 as well.

For the reasons give above, Appellants respectfully submit that the rejections should be withdrawn and the claims should be allowed.

Respectfully submitted,



John C. Freeman
Registration No. 34,483
Attorney for Appellants

BRINKS HOFER
GILSON & LIONE
P.O. Box 10395
Chicago, Illinois 60610
(312) 321-4200

Dated: November 13, 2006

VIII. CLAIMS APPENDIX

22. A system for displaying information about an insurance claim for an insured event, the system comprising:

 a server component including an event processor and a task engine application program that interacts with the event processor to enable said insurance claim to be processed; and

 a data component residing on the server component, the data component comprising a claim folder that decomposes a claim related to the insured event into a plurality of levels, the plurality of levels including a policy level, a claim level, a participant level and a line level,

 wherein the server component is configured to generate a user interactive interface that interactively displays at least one of the plurality of levels reflecting information related to a policy, the claim, claimants and an insured person in a structured format to a plurality of users, and to allow each of the users to simultaneously interact with one of the plurality of levels to retrieve and enter data for the same insurance claim;

 wherein the event processor maintains clear encapsulation of responsibilities of said system for displaying information from said event processor, wherein said responsibilities do not include functions performed by said event processor, interacts with the data component to identify a data event that affects data in the claim folder, determines a response, identifies a system component to enable said claim to be processed and transmits the data event to the identified system component;

wherein when said identified system component is the task engine, the task engine evaluates the data event, determines claim characteristics and matches the characteristics to tasks to automatically generate a list of tasks to be taken by one of the plurality of users handling said insurance claim to direct a workflow for said insurance claim to be processed.

41. The system of Claim 22 wherein the server component displays policy level information comprising information related to covered autos for auto claims, information related to covered property for property claims and information related to covered yachts for marine claims.

42. The system of Claim 22 wherein the server component displays claim level information comprising details information, facts of loss information, events information and liability information.

43. The system of Claim 22 wherein the server component displays participant level information comprising details information and contact information, information related to the insured event, injury information and disability management information.

44. The system of Claim 22 wherein the server component displays line level information comprising information related to damaged vehicles for vehicle lines, information related to damaged property for property lines and information related to damaged yachts for marine lines, and information related to the insured events, damages and negotiation associated with the vehicles, property and yachts.

45. The system of Claim 22, wherein the server component displays claim level information comprising details information, facts of loss information, events information and liability information.

46. The system of Claim 22 wherein the server component displays participant level information comprising information related to persons involved in the claim, information related to the role of persons in the claim and contact information of the persons.

47. The system of Claim 22 wherein the line level comprises a user interface enabling the capture of negotiation information.

48. The system of Claim 22 further comprising a client component in communication with the server component, wherein the client component is configured to provide information concerning an individual in the insured event and for allowing one of the plurality of users to link the individual to the insured event.

49. The system of Claim 48 wherein the client component is configured to display the user interface as a response to the communication with the server component.

50. The system of Claim 49 wherein the client component is configured to allow one of the plurality of users to edit information associated with the plurality of levels.

51. The system of Claim 49 wherein the data component is configured to allow one of the plurality of users to search for information associated with one of the policy level, the claim level, the participant level and the line level.

52. The system of Claim 22 wherein the server component displays participant level information comprising a category of historical information, a claim index and contact information.

53. The system of Claim 22 wherein the server component displays policy level information comprising information on the claimants that are injured with disabilities.

54. The system of Claim 22 wherein the server component displays policy level information comprising specific information on injuries suffered by the claimants.

55. The system of Claim 54 further comprising a statistical model for claim practices and risk selection that uses the specific information on injuries suffered by the claimants, said specific information being stored in claims folders and accessible by said server component.

56. The system of Claim 54 wherein the specific information is represented by ICD-9 code.

57. The system of Claim 22 further comprising a client component in communication with said server component, said client component displaying a claim tree associated with said policy.

58. The system of Claim 57 wherein said claim tree lists said policy, said insured person, claimants and related lines in a claim tree format.

59. The system of Claim 57 wherein said client component further displays claim level tabs.

60. The system of Claim 57 wherein said claim folder displayed on a client component can be changed between a view mode and an edit mode.

61. The system of Claim 60 further comprising menu options displayed on said client components, wherein said menu options depend upon whether said claim folder is in said view mode or said edit mode.

62. The system of Claim 22 wherein said event processor further provides a task due date for each task of said list of tasks.

63. The system of Claim 22 wherein said event processor further records tasks completed for every claim.

64. The system of Claim 22 wherein said event processor generates a historical record for any task which is determined to be completed related to the processing of said claim.

65. The system of claim 57 wherein the client component uses an optimistic locking mechanism when the claim tree is displayed to the plurality of users simultaneously.

66. A system that displays insurance claim information about an insured event, the system comprising:

- an event processor that identifies a data event, determines a response, identifies a system component to process an insurance claim and transmits information regarding the data event to the identified system component;
- a task engine application program that interacts with the event processor to enable the insurance claim to be processed;
- a data component comprising a claim folder that decomposes a claim related to the insured event into a plurality of levels, the plurality of levels including a policy level, a claim level, a participant level and a line level; and
- a user interactive interface that is generated by a server that interactively displays information from at least one of the plurality of levels in a structured format to a plurality of users, allowing each of the users to simultaneously interact with one of the plurality of levels to retrieve and enter data for the same insurance claim, the entered data triggering the data event,
 - wherein when the event processor identifies the task engine as the system component to process the insurance claim, the task engine evaluates the event, determines claim characteristics for the event and matches the characteristics to tasks to automatically generate a list of tasks to be taken by one of the plurality of users handling the insurance claim to direct a workflow for the insurance claim to be processed.

67. A system that displays insurance claim information comprising:

a data component that includes a claim folder that decomposes a claim related to an insured event into a plurality of levels, the plurality of levels include a policy level, a claim level, a participant level and a line level;

a user interactive interface that is generated and interactively displays information from at least one of the plurality of levels in a structured format to a plurality of users, wherein a plurality of users via a plurality of interfaces is allowed to simultaneously interact with one of the plurality of levels to retrieve and enter data on the same insurance claim; and

an event processor that identifies the entered data as a data event, determines a response for the data event and identifies a system component to process the response and transmits information for processing the claim to the identified system component.

IX. EVIDENCE APPENDIX

None.

X. RELATED PROCEEDINGS APPENDIX

None.